Workshop timeline with 5 phases[Skip to current tasks](https://my.uopeople.edu/mod/workshop/view.php?id=142262#mod_workshop-userplancurrenttasks)

**Setup phase**

**Submission phase**Current phase

* Task to do[Submit your work](https://my.uopeople.edu/mod/workshop/submission.php?cmid=142262&id)
* Task infoOpen for submissions from Wednesday, 18 April 2018, 11:05 PM (today)
* Task infoSubmissions deadline: Wednesday, 25 April 2018, 10:55 PM (7 days left)

**Assessment phase**

* Task infoOpen for assessment from Wednesday, 25 April 2018, 11:05 PM (7 days left)
* Task infoAssessment deadline: Wednesday, 2 May 2018, 10:55 PM (14 days left)

**Grading evaluation phase**

**Closed**

[Instructions for submission](https://my.uopeople.edu/mod/workshop/view.php?id=142262)

For the unit 2 assignment, you will develop a program that will simulate the rolling of a pair of dice. Your program must simulate rolling two dice and keep track of how many rolls are required before snake eyes are rolled (Note: “Snake Eyes” means that both dice show a value of 1). You will need to use a loop structure to repeatedly ‘roll the dice’. You will also need to make use of the if statement to determine if both die show a 1. You can use the example of an if statement below in your program and you can also read ahead to next week's lesson for more information on the if statement.

To roll the dice, you can use the following Java statement which generates a random number between 1 and 6 (to represent the six sides of the dice).

die1 = (int)(Math.random()\*6)+1

This statement has some interesting characteristics that need to be explained. The first thing we notice is that we are assigning something to a variable with the name die1. You will need to execute this statement twice, once for a variable die1 and the second time to assign to variable die2 to represent the roll of two dice.

The second thing we notice about this is this weird (int) placed in front of the parentheses that contain a subroutine call and a math expression (Math.random()\*6).

What the (int) in this case is doing is called a cast. The cast is a way to force java to convert data from one data type to another.

The subroutine (actually this is calling the class Math and the method random) Math.random() will return a number that is between 0 and 1. This number is a floating point number.

By multiplying it by the number 6 we will get a random number between 0 and 5. This number will be a floating point number and is likely to be a fraction meaning that it will have digits to the right of the decimal point.

What the cast or (int) in this case does is convert this number into an integer. As you recall from earlier discussions, this will truncate all the digits to the right of the decimal point. For example assume that the Math.random() subroutine returns the value of .523432. This value is then multiplied by 6 resulting in 3.14059 to this we add 1 resulting in 4.14059. The reason that we are adding a 1 is that we want a number between 1 and 6 starting at 1. By adding a 1 we know that the starting number will always be 1 or greater.

What the (int) will do is to truncate everything to the right of the decimal point and then assign it to the variable die1. The resulting value in die1, in this case, would be 4.

An if statement that will test for ‘snake eyes’ would be as follows.

if (die1 == 1 && die2 == 1) {

// do something

}}

You must complete your program, test, debug, and execute it. You must submit your java code file. Post the output of your program by specifying the number of rolls of the dice before you got snake eyes. Run the program at least five times and report your results.

**Challenge:** See if you can modify your program using a for loop to simulate running your program 1000 times and calculate the average number of rolls required to get snake eyes … what is the average number of rolls?

**Helpful Hints**

When using the Netbeans IDE, there are issues with package naming that can be a bit confusing. To help you understand how to set up your environment to use the TextIO.java library. You need to do the following:

1. Download the TextIO.java to your local computer or device. If you are using the Program service you need to add TextIO.java as a file in your project and your programs should rook If you are using Netbeans then we have a bit more work to do.
2. Create a project for your assignment. Make sure to select category Java and project Java Application. When you create your project it will create a package that has the name of your project. The following screen shot shows how this should look. You will have a source package with the name of your project and in it a Main.java file into which you put your code. You should make sure that your Main.java file looks similar to example below.  
     
   see image [Hint1]
3. You must upload your TextIO.java file and save it under the default package in the source packages of your project. You can see what this looks like in the preceding figure in the left side of the screen where under the project “SnakeEyesCount” you have Source Packages and within this “<default packages>” and the package “snakeeyescount”.

See image hint2

1. Finally you must edit your TextIO.java source code file and add a line at the top of the file that has “package <nameofpackage>”. In the example below the name of the package is snakeeyescount. Add the proper package line at the top of TextIO.java